Code worthy

The humble bar code has changed the face of retailing forever, but what purpose does it serve and how does it work? Mike Bedford explains

A revolution has taken place in the retail trade over the last decade. Nine years ago I was amazed when I encountered bar code scanning at a supermarket checkout in Seattle. Such technology was quite unheard of in the UK at that time.

In pre-bar code days, every item on the supermarket shelves had to be individually marked with a price sticker. Today, all products, as they come from the manufacturer, have bar codes printed on them. These codes don't specify the price, but they do uniquely identify the product. The actual price is recorded in only two places: on the shelf in plain English, and in a computer database which is interfaced to the checkout tills. So when a purchase is passed across the scanner, the number 9770955857028 (for example) is read from that item's bar code, and the computer recognises that the price of this commodity is £1.19 and that the description of the item to be printed on the receipt is "Computer Shopper".

The result is increased efficiency at the checkout, greater throughput, lower labour costs and shorter queues. And by linking into a stocking system, an up-to-the-minute record of stock can be made available, triggering re-ordering immediately it's required. Retailers implementing such a system have seen immediate benefits in terms of improved stocking levels.

The methods

Generally, anything that moves around and requires monitoring at one or more points on its journey is a candidate for some type of automatic identification. The list of applications includes warehousing, manufacturing, retailing, parcel and baggage tracking, credit card handling, cheque recognition, postal services, toll road/bridge billing, physical access control, computer access control, workplace clocking and libraries. Of these, the first four nearly always involve bar coding, and some of the others may use bar codes in preference to other technologies.

Having raised the topic of alternative automatic identification technologies, we need to consider the choices available and the factors which might influence the decision to use bar coding. The main methods of automatic identification are described below. Other methods also exist, but tend to be more specialised in their application.

● Bar codes

In essence a bar code is a series of lines of varying widths separated by gaps of varying widths. It's printed using conventional printing techniques and read optically.

● Magnetic stripes

A magnetic stripe is a piece of magnetic material, very much like a piece of computer or audio tape, stuck to some rigid surface. This is both written and read by something similar in principle to a tape drive, the main difference being that the stripe is passed through it by hand. The brown stripe on the back of a credit card or a London Underground ticket is a magnetic stripe.

● OCR

Optical character recognition is the reading by machine of characters intended to be read by eye. Many OCR systems can cope with any well printed characters, but older examples of the technology (for example, the identification number on the bottom of a cheque) require specially designed OCR characters.

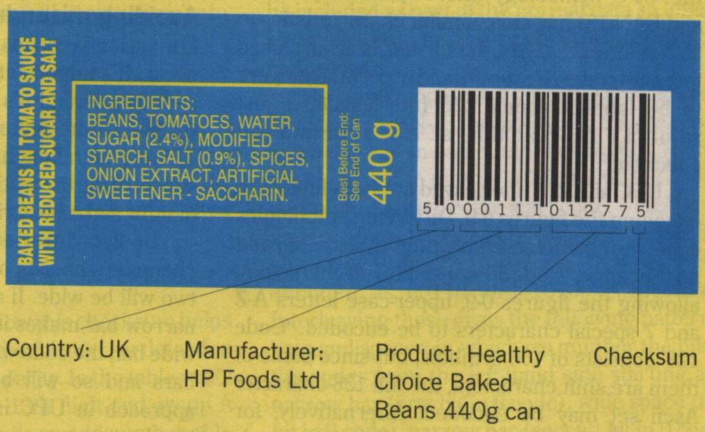
● RF

RF automatic identification involves the transmission of a digitally encoded radio signal in response to an enquiry transmission from the interrogating equipment. This is a proposed method of charging for the use of cars in city centres and on toll roads.

Pluses and minuses

All of these techniques have their pros and cons. RF is certainly the most expensive. The major

Never the same bar twice


BAKED BEANS IN TOMATO SAUCE WITH REDUCED SUGAR AND SALT
INGREDIENTS: BEANS, TOMATOES, WATER, SUGAR (2.4%), MODIFIED STARCH, SALT (0.9%), SPICES, ONION EXTRACT, ARTIFICIAL SWEETENER - SACCHARIN.
Best Before End: See End of Can
440 g
5 000111 012775
Country: UK
Manufacturer: HP Foods Ltd
Product: Healthy Choice Baked Beans 440g can
Checksum

The Studio

The EAN is an international organisation which has the task of allocating blocks of the 10 billion (American billions, that is – 10 million million) possible EAN-13 codes to various national bodies. The way this has been done is to take the first two digits as national codes, allowing bodies in each of up to 100 countries to allocate sub-blocks to individual companies from their own blocks of 100 thousand million. The code for the UK is 50, and the British association which administers the codes is the ANA, the Article Numbering Association. Following the two national digits, the next five digits define the manufacturer and the following five digits define the actual item. As explained in the text, the one remaining digit is a checksum. What this means is that in each country, blocks of codes can be allocated to up to 100,000 companies, each of whom can have 100,000 different products.

We should also mention the smaller EAN-8 symbol, which can be found on some retail goods. This is really intended for items which are too small to be labelled with an EAN-13 symbol, and because there are far fewer 8-digit codes than 13-digit ones (100,000 times fewer, to be precise) these are now only allocated if the need can be proved. Looking around, there do seem to be a few large items carrying EAN-8 bar codes which must have somehow slipped through the net.

The astute will by now have realised that the bar code on *Computer Shopper*, which we've used as an example of EAN-13 encoding, doesn't fit this pattern, having as its first two digits 97 instead of 50. Books and magazines are, indeed, exceptions. All books have bar-coded symbols which start with 978, and all magazines have symbols which start with 977. Following this is the ISBN or ISSN number respectively, and magazines have an extra add-on pair of digits to identify the issue number.

➧

advantage is that the technology can operate over a reasonable distance and can cope with fast-moving objects. It also shares an advantage with magnetic strip identification, namely that the result of the interrogation isn't fixed. In a multiple-journey travel ticket, for example, the number of journeys remaining has to be decremented each time a journey is made. A read/write scheme, as opposed to a read-only mechanism, is therefore essential.

OCR and bar coding share one very great advantage: they require no special technology to place the symbol on the object being coded. OCR characters or a bar code can be printed using conventional techniques. Turning to the reading process, the main difference between bar codes and ordinary characters is that a bar code can be scanned in a single pass, whereas OCR characters need multiple passes at different vertical positions. The greater complexity of an alphanumeric character also means that its machine reading is more error-prone than that of bar codes. It also implies that bar code scanning is the least expensive technology. On the other side of the coin, ordinary printing can be read by eye and doesn't take up as much space as a bar code carrying the same information. To make them visually readable, bar codes frequently have the alphanumeric representation printed below them.

Propping up the bar

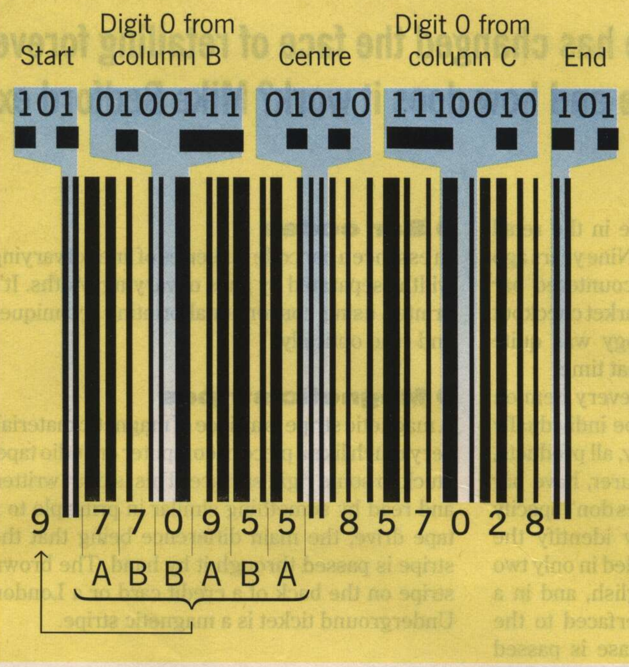
So what exactly is a bar code? The first thing to point out is that there are numerous different coding schemes in regular use. Although there are some pros and cons which would point to the use of one symbology in preference to another for a particular application, a lot comes down to following historical guidelines. So, for example, the 'UPC' symbology (more specifically EAN-8 and EAN-13) has quite similar characteristics to '2/5 Interleaved', yet the former is always used for retail codes. If a new brand of cornflakes were to be labelled using 2/5 Interleaved, it just wouldn't be readable at the supermarket checkout.

UPC and 2/5 Interleaved are both limited to encoding the figures 0-9. Different symbologies cope with character sets of different sizes. Moving up the range, 'Code 39' consists of 43 characters, allowing the figures 0-9, upper-case letters A-Z and 7 special characters to be encoded. 'Code 128' consists of 106 symbols, but since some of them are shift characters, the full 128-character Ascii set may be encoded. Alternatively, for numbers only, each symbol can represent a pair of figures. The 'Telepen' code also provides the full Ascii character set, but without resorting to the use of shifts.

In selecting symbologies, the basic rule is that a code should be chosen which supports only the characters required. To understand why this is so, we need to consider how some codes manage to handle more types of characters than others. The code either has to be made longer (ie more stripes per character) or a greater number of different bar widths have to be allowed. Making the code longer means the bar codes take up more room; increasing the number of bar widths means the scanning equipment has to be more accurate and/or misreads will be more frequent. Clearly, moving in either of these directions is best avoided.

A final reason for the choice of one symbology over another is concerned with the printing technology. In general, as we've already seen, bar codes can be printed using conventional methods. However, accuracy requirements do vary from one coding scheme to another, and this may impose some restriction on the choice of printing method for a given symbology.

EAN-13 encoding



Avoiding misunderstandings

For various reasons, read errors will inevitably occur. It's important, however, that when a bar code is misread it's recognised as such rather than being interpreted as a different valid number. This is generally achieved by using only a certain subset of the theoretically possible combinations of light and dark stripes.

In 2/5 Interleaved, for example, each character consists of five bars of which exactly two will be wide. If a speck of dirt adjacent to a narrow bar makes it appear to the scanner as a wide bar, the code will appear to have three wide bars and so will be rejected as invalid. The approach in UPC is a more conventional one used in many systems which involve the transmission of binary data. Here, characters from each column have an associated 'parity'. If a column has odd parity then all codes have an odd number of 1s; if a column has even parity then only codes with an even number of 1s are used. Once again, small errors will be detected – although, as in all parity systems, a severe read error affecting more than one bit can go undetected.

In EAN-13, a second method of error detection is employed to make the system even more secure. Character number 1 (ie the first character counting from the right) contains no real data but is instead a 'checksum' digit. When the bar code is printed, the value of the checksum is calculated from the values of the other 12 figures (as described in the section on EAN-13 encoding). On scanning the bar code, the reader calculates the expected checksum from characters 2 to 13 and compares this with the first character as actually scanned. Any discrepancy indicates a read error. Another fundamental design requirement of bar code symbologies is that it shouldn't be possible to read codes in the wrong direction and in so doing get a different result. 2/5 Interleaved uses different start and stop characters to eliminate this potential problem. EAN-13, on the other hand, uses the same characters for start and stop (and also something very similar to mark the middle point), but ensures that the symbol can't be misinterpreted by employing the A and B codes in the left-hand side and the C code in the othe[illegible] half.

Since EAN-13 has the same pattern for the start and the stop code, it would be appropriate to ask why they're there at all. This brings out one final design requirement. The bar code scanning method we see at supermarket checkouts involves passing a laser beam over the symbol at a constant speed. It's a requirement of bar codes, however, that they can be read by a simple handheld scanner in which the scan speed will vary from one scan to the next. It's essential, therefore, that a symbol starts with something which is predictable, as this can then be used to deduce the scanning speed and so interpret the remaining bars correctly. If a known pattern is placed at both ends, the symbol becomes readable in either direction, and a more accurate speed estimate can be achieved by averaging the sta[illegible] and end speeds.

EAN-13

EAN-13 is a specific example of the UPC code, the other main variant being EAN-8. These are the two codes used in retail applications, and both allow the encoding of figures only. The number 13 refers to the number of figures encoded, but things aren't quite that simple. Firstly, the left-hand digit isn't explicitly encoded

EAN-13 encoding table

Digit	Group A	Group B	Group C	13th digit
0	0001101	0100111	1110010	AAAAAA
1	0011001	0110011	1100110	AABABB
2	0010011	0011011	1101100	AABBAB
3	0111101	0100001	1000010	AABBBA
4	0100011	0011101	1011100	ABAABB
5	0110001	0111001	1001110	ABBAAB
6	0101111	0000101	1010000	ABBBAA
7	0111011	0010001	1000100	ABABAB
8	0110111	0001001	1001000	ABABBA
9	0001011	0010111	1110100	ABBABA

➧

in the bars but is deduced from the way the reamining six left-hand digits are encoded; and secondly, the right-hand digit is a checksum rather than true data. Basically, an EAN-13 symbol comprises the following components from left to right: a start code, the most significant six figures, a centre code, the least significant six figures, and the end code.

For the purpose of this explanation, we'll consider the encoding of the bar code on the front of the copy of *Shopper* you're currently reading. In fact this bar code is 'EAN-13 add on 2', which means it's a standard EAN-13 symbol with an extra two digits that (in this case) are used to specify the issue number. In this example, however, we'll ignore this extra part, leaving us with just the main body of the code to consider.

The internationally assigned number for *Computer Shopper* magazine is 9770955857028. The first digit, the 9, isn't associated with a particular portion of the symbol but instead dictates the way the next six digits are encoded. In the table (see below), the right-hand column refers to the encoding of the left-hand digit. The code for 9 is ABBABA, which means that the next six figures are encoded from columns A, B, B, A, B and A of the table respectively. Before we start on these, however, we need to include the start pattern 101 which, in accordance with the rules of encoding this symbol, means a single-width bar followed by a single-width space and another single-width bar.

Now to the second most significant figure, the 7, which, as we've already seen, must be encoded from column A. The binary representation is therefore 0111011, which in bar terms is represented by a single-width space, a triple-width bar, a single-width space and a double-width bar. The next figure is another 7, but this is now encoded from column B, giving 0010001 – in other words, a double-width space, a single-width bar, a triple-width space and a single-width bar. Things continue in this way for the figures 0, 9, 9 and 5.

We're now halfway through the symbol, so the centre pattern 01010 must be inserted. The remaining six digits are now encoded from column C of the table, which means that 8 is 1001000, and this proceeds until the final 8 has been represented. To complete the symbol another start/stop pattern, 101, is added.

So far we've considered the final character, the 8, to be data like all the others and, as far as the conversion to bars is concerned, this is indeed so. But the purpose of this right-hand digit is for error detection, and is defined as a modulo 10:31 checksum. It's calculated from the remaining 12 figures in the following manner. In this explanation, we adopt the formal numbering scheme in which the right-hand digit (ie the checksum) is figure 1 and the left-hand digit (ie the most significant) is figure 13. The example worked through is that of the *Shopper* bar code illustrated here.

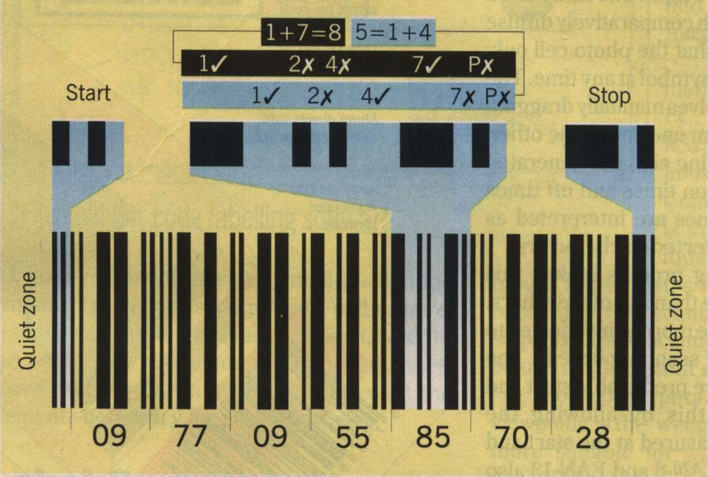
2 of 5 encoding table

Digit	Weighted position				
	1	2	4	7	P
0	0	0	1	1	0
1	1	0	0	0	1
2	0	1	0	0	1
3	1	1	0	0	0
4	0	0	1	0	1
5	1	0	1	0	0
6	0	1	1	0	0
7	0	0	0	1	1
8	1	0	0	1	0
9	0	1	0	1	0

❶ Add together all the even digits (ie second, fourth, sixth and so on):
2+7+8+5+0+7=29

❷ Multiply the result of ❶ by 3:
29x3=87

2 of 5 Interleaved encoding



❸ Add all the odd digits together (excluding, of course, the first (right-hand) digit, which we're in the process of calculating):
0+5+9+7+9=35

❹ Add the result of ❷ to the result of ❸:
87+35=122

❺ The checksum is the result of ❹ subtracted from the next highest multiple of 10:
130–122=8

We've already seen how this checksum helps in error detection. The other bit of error detection designed into EAN-13 is parity. In the table you'll see that all codes in the 13th digit and group A columns have odd parity (ie an odd number of 1s or Bs), whereas those in groups B and C have even parity.

2 of 5 Interleaved

The 2 of 5 Interleaved coding scheme also handles numeric data only, but is not fixed-length and so can cope with any even number of digits (an odd number of digits can be encoded by adding a leading zero in the most significant position). In simple terms, a 2 of 5 Interleaved symbol consists of a start pattern at the left-hand edge, any number of character pairs, and an end pattern at the right-hand edge. The start and end patterns are different.

A character pair consists of five bars and five spaces, in which the widths of the bars represent the most significant digit and the widths of the spaces represent the least significant digit. This is the meaning of the term 'interleaved' in the name of this symbology. The table shows the coding scheme, which follows a pseudo-BCD rationale. Except for zero, all characters can be encoded by placing a 1 in those positions in the table which will add up to that figure. So, for example, 8 is equal to 1+7, so the code for this digit is the code with 1s in the 1 and 7 positions: 1001. A parity bit is also added for integrity checking. This makes the total number of 1s in each code up to two, so the full code for 8 is actually 10010.

Since all codes have exactly two out of the five bits set, the code is called 2 of 5 Interleaved. This explains, of course, why true BCD – in which the weighted values would be 1, 2, 4 and 8 – wasn't used. The code for 7 in this case would be 1110, and this would break the 2 of 5 rule. In turning the code into bars or spaces, the rules are that a bar or a space for a 1 is between two and three times as wide as the corresponding one for a 0.

The standard doesn't specify a checksum character, although there is inherent parity checking in each character. Nevertheless, a checksum can be included as part of the normal data so long as its inclusion is agreed upon by all the parties concerned.

As our example, we'll take the same figure as used in the EAN-13 example – the *Shopper* code. Since there is an odd number of digits, a leading zero is added to give 09770955857028. The first part of the symbol is the start code, which is the fixed pattern shown. Following this is the first character pair, 09, these two figures being encoded as 00110 and 10100 respectively. Interleaving these gives the following pattern of bars and spaces, in which we number the bits in the codes from the left-hand side starting at 1: narrow bar (first bit in 0 code), wide space (first bit in 9 code), narrow bar (second bit in 0 code), narrow space (second bit in 9 code), wide bar (third bit in 0 code) and so on, giving narrow space, wide bar, wide space, narrow bar, narrow space. The second character pairs are encoded in a similar manner. Finally, the end pattern completes the symbol.

The symbol we have just constructed is actually called an ITF-14 bar code and, like EAN-13, is used for retail commodities. Unlike EAN-13, however, an ITF-14 symbol will never appear on the item that is actually placed on the shelf at your local supermarket. Instead it is used to identify the contents of the packing cases and boxes which will only get as far as the store's warehouse. ➧

The technology

In general terms, a bar code is read by shining a light onto it and detecting what's reflected back from various positions across the width of the symbol. The black areas (ie the bars) reflect very little light, whereas light will be reflected from the light areas. The software on the machine to which the scanner is attached thus receives an on/off signal as the code is scanned, and converts this to numbers or Ascii characters using the decoding rules for the symbology in question.

The simplest and also the cheapest type of bar code reader is the pen or wand. To avoid confusion with the lightpens used as pointing devices on monitors, the word 'wand' is preferred. In the tip of the wand are two types of optoelectronic device: a number of LEDs which illuminate the symbol, and a photosensitive cell which detects the reflected light. The LEDs flood an area of the bar code with comparatively diffuse light, but lenses ensure that the photo cell only 'sees' a single spot on the symbol at any time. The operation of the wand involves manually dragging it across the bar code from one end to the other. As this is done, a fluctuating signal is generated and the durations of the on times and off times are measured. These times are interpreted as bar widths and thus converted to characters.

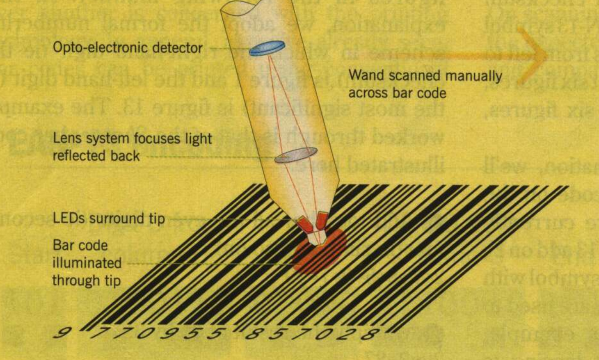
The manual scanning process makes this method more error-prone than any of the others. The software also has to be more sophisticated to deal with a wide range of scanning speeds. The fact that all bar codes have predictable start and stop characters eases this by allowing the scanning speed to be measured at the start and the end of the process. EAN-8 and EAN-13 also have a predictable pattern in the middle, and so can cope better with a varying scan speed, the software being able to work out an average speed for each half of the symbol. Figure 1 shows the basic design of a bar code wand.

A variant which is conceptually similar, although of a different mechanical appearance, is the slot reader. This allows a paper or card containing a bar-coded symbol to be read by manually passing it through the slot. This type of reader is typically used to read bar-coded cards for security purposes and could be used, for example, to unlock automatic doors.

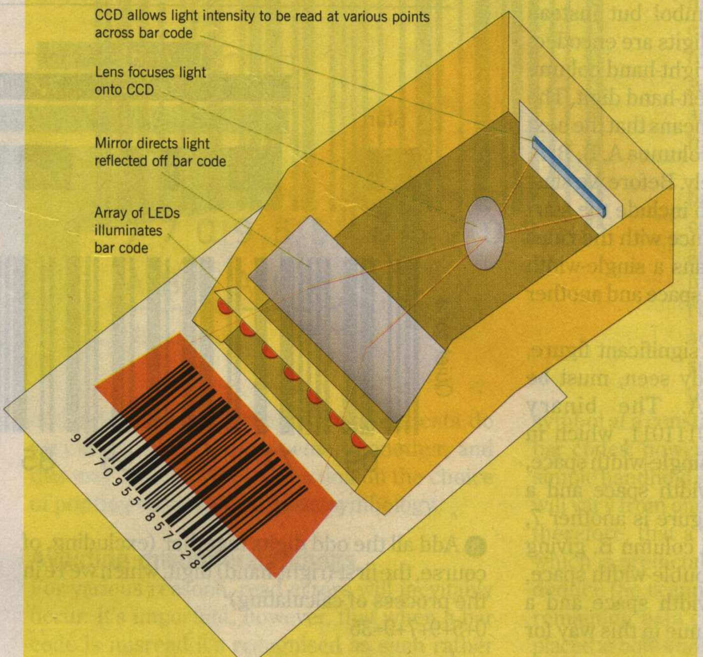
The next simplest solution is the CCD scanner. This is also a handheld device, but one which is placed over the entire symbol rather than being manually tracked across it. Figure 2 illustrates a CCD scanner. An array of LEDs illuminates the whole bar code. Light reflecting from the symbol passes into the scanner via a narrow aperture and is focused onto the surface of a CCD (charged coupled device), an optoelectronic component which is also used in TV cameras. The driving electronics is able to address any point across its width, and so determine the intensity of the light being reflected from any position across the bar. The CCD is therefore scanned electronically by the software.

Light fantastic

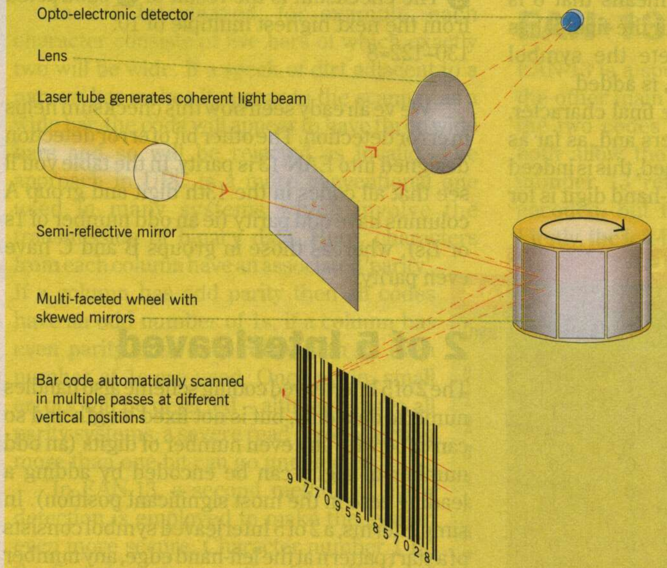
The laser scanner is the most sophisticated and most flexible means of reading bar codes, and is the type most commonly found in supermarkets.


Opto-electronic detector
Wand scanned manually across bar code
Lens system focuses light reflected back
LEDs surround tip
Bar code illuminated through tip
9 770955 857028

① The wand is the simplest kind of bar code reader


CCD allows light intensity to be read at various points across bar code
Lens focuses light onto CCD
Mirror directs light reflected off bar code
Array of LEDs illuminates bar code
9 770955 857028

② The CCD scanner reads the whole bar code at once


Opto-electronic detector
Lens
Laser tube generates coherent light beam
Semi-reflective mirror
Multi-faceted wheel with skewed mirrors
Bar code automatically scanned in multiple passes at different vertical positions
9 770955 857028

③ The laser scanner is the most sophisticated reader

The major advantages are that manual scanning is not required, accurate registration between scanner and symbol isn't necessary, and – within limits – the bar code can be read at a distance and at any orientation. It's only necessary for the operator to place the bar coded item close to the scanner with the bar code facing and approximately parallel to it. The item doesn't even need to be still, and so can be swept past.

This sounds very clever and, as we shall see, it's all done with mirrors. Figure 3 shows the basic principles. The light is generated by either a conventional helium-neon laser or a laser diode and, being a laser beam, i relatively non-divergent. Th means that even at a distance it will illuminate just a single spot on the bar code. Between the laser generator and the bar code, however, is a multi-faceted rotating wheel with mirrors on each face. There's also a semi-reflective mirror, but we can forget about this for now, since the laser beam will just pass straight through it in the outward direction. As the wheel of mirrors rotates, the beam is reflected at a constantly varying angle as each mirror goes past. This has the effect of scanning the complete symbol (and the space to either side of it) in multiple passes. If the mirrors are placed non-parallel to th axis of the wheel, subsequent passes will be made through the symbol at slightly different horizontal positions. This eliminates problems due to any imperfection on the bar code at a single horizontal position.

The one snag with this approach is that it will only work if the bar code is in the correct orientation. This is quite acceptable for warehousing applications, where boxes pass the scanner on a conveyor belt and will all be similarly orientated, but places an undue restriction on supermarket checkout operators. To overcome this limitation, the arrangement shown in Figure 3 is augmented with a second reflective multi-faceted wheel at right angles to the first. We won't illustrate this form of scanner, since it gets difficult to see exactly what's happening, but let's just say that this solves the problem.

So far we've described a means that will scan a bar code at any angle with a laser beam. If the laser beam hits the bar code exactly at right-angles, the reflected beam will follow exactly the same path as the incoming beam. When it hits the semi-reflective mirror, some of the beam will be reflected onto the light sensor. Normally, of course, the laser beam won't hit the bar code at right angles, so the reflected beam will follow a different path. Different possible return paths are shown on the figure as dotted lines. So long as the light hits any part of the semi-reflective mirror, the lens assembly will cause the beam to reach the sensor, allowing the light intensity to be read.

Laser scanners can be either fixed or handheld. The fixed variety is used in supermarkets and in warehousing applications to track objects on conveyor belts. Handheld ones are used in a similar way to the CCD scanner, the major difference being that a laser scanner doesn't need to be as close to the symbol and so can be used in a 'point-and-shoot' manner.

The products

We now turn our attention to practical products which will allow you to print and read bar codes using a PC. Although in the case of bar code printing we'll look at some real packages, we're not setting out to present a comparative review; instead we'll consider a couple of products from one manufacturer which are typical of much of the software available. Though you may like to consider other options, both of the products reviewed here do exactly what they claim and would be entirely suitable for anyone moving into bar code labelling.

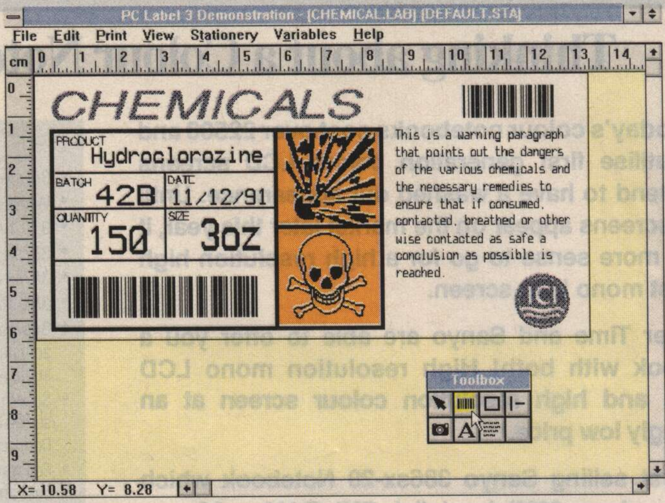
For a comprehensive list of companies specialising in automatic identification products, call AIM UK, the automatic identification industry's trade association (phone number below).

Overdrive

Dedicated bar code printers do exist. To print the EAN-13 bar code 977095857028 on such a printer, you might send the code:

```
<ESC>it5 b977095857028\
```

The printer contains all the intelligence necessary to generate the symbol, so it could be driven from any standard software. Unfortunately, bar code printers are expensive. However, *Overdrive* from Image allows any Epson-compatible dot matrix or H-P-compatible laser printer to emulate the capabilities of a dedicated bar code printer. It operates as a memory resident utility, monitoring the output to the printer port, extracting any bar code commands and converting them to a string of the printer's graphics commands to generate that symbol. This software is therefore an ideal package for outputting bar codes from standard software you already have. It will also allow you to write your own applications without having to get involved in the intricacies of bar code printing. *Overdrive* handles Code 39, Interleaved 2 of 5, Industrial 2 of 5, EAN-8, EAN-13, UPC-A, UPC-E, Telpen, Telpen Numeric, Codabar, MSI (11/11 or 10/11 checksum) and Code 128 (sets A, B and C) symbologies. It provides control over bar width ratio, symbol height, human-readable characters below the symbol, variable number of print passes (for dot matrix), symbol position and symbol rotation (for laser printers). Furthermore, in addition to bar code printing facilities, *Overdrive* can print large alphanumeric characters, lines, boxes and blocks. In short, it handles all the elements necessary for printing labels.



CHEMICALS

PRODUCT Hydroclorozine

BATCH 42B

DATE 11/12/91

QUANTITY 150

SIZE 3oz

This is a warning paragraph that points out the dangers of the various chemicals and the necessary remedies to ensure that if consumed, contacted, breathed or other wise contacted as safe a conclusion as possible is reached.

④ Typical bar code labelling software

Label Designer

This package offers very much the same sort of facilities as *Overdrive*, but instead of being command-driven it has a menuing front end for the design of labels. A label is defined by filling in a table in which each line defines the contents and characteristics of fields on the label. Fields may be lines, boxes, text or bar codes. The lines and boxes are used to split up the whole label into areas which can then take text or bar codes. The text can be in a number of sizes, and there are various options for the bar codes. The number represented by the bar code can be made to auto-increment for serial numbering.

Of course, there are facilities to define just how the labels are to be arranged on the page. Unfortunately, however, there's no label preview mode, so designing a new label is a matter of trial and error. For an organisation printing only a few different types of label this is probably acceptable, but if you frequently need to create new designs I feel this approach would get a bit tedious.

Image does also have *PC-Label*, a wysiwyg *Windows 3* label designer which overcomes the drawbacks of the Dos program. Unfortunately I was only provided with a demonstration disk of this package, so I wasn't able to do a full review. On the face of it, though, *PC-Label* does appear to be a user friendly package and has the bonus of allowing bitmapped graphics to be imported onto the labels. Figure 4 shows this product in action. A fourth product available from Image is *Publisher's Barcode*. This allows you to create bar codes in .PCX format for subsequent insertion into word processed or DTP documents.

Scanners

We've already described the basic principles of a number of bar code readers. One common denominator is that they output an on/off electrical signal corresponding to the light and dark areas on the symbol. This is then pre-processed by low-level software to convert the electrical signals into recognisable figures or letters. Most bar code scanners for the PC have their own built-in processors which carry out this task, freeing the PC from this chore. They therefore plug into the PC and input normal Ascii data.

There are two common methods of interfacing to the PC. The first of these uses RS232, so the scanner plugs into one of the PC's serial ports, Com1 or Com2. The drawback is that most software takes its input from the keyboard or mouse rather than a serial port. This therefore precludes the scanner's use with standard software. RS232 scanners will therefore normally be used with custom software.

The second type of scanner, from a connection viewpoint, is the 'wedge' scanner, which is much more suitable for use with off-the-shelf software packages. A wedge scanner fits between the keyboard and the PC and simulates input from the keyboard, which is, of course, what most software expects. Since the keyboard is still there, however, the user can switch between keyboard and scanner input. These scanners usually allow a range of characters to be added before or after the symbol to suit the software. So, for example, by selecting Enter as a terminator, the need to press Enter on the keyboard after reading each bar code can be obviated. ■

Contacts

Overdrive £245
Label Designer £295
PC-Label £595
Publisher's Barcode £175
Image Computer Systems
(0202) 876064

Bar code reading equipment
Datalogic
(0582) 794000

Bar coding books; technical information; directory of UK automatic identification companies
AIM UK
(0422) 359161
(Trade association for the automatic identification industry)

Retail bar codes
ANA (071) 836 3398
(British association responsible for the allocation of bar codes)

Thanks to Richard Adlington of Datalogic UK for kindly providing information on bar code scanning techniques.

The Amazing Sanyo Notebook Is Now Available with Colour for Just £1299

3 YEAR
3
EC WIDE
WARRANTY

Thinking about a Colour Notebook?

Today's colour notebooks cost over £2500 and utilise first generation colour LCD screens which tend to have a washed out appearance. Until better screens appear on the market later this year, it makes more sense to go for a high resolution high contrast mono LCD screen.

However Time and Sanyo are able to offer you a notebook with both! High resolution mono LCD screen and high resolution colour screen at an amazingly low price.

The best selling Sanyo 386sx-20 Notebook which features a large 60Mb hard disk, 3Mb RAM and 3 year warranty is now available with an external 14" Panasync Super-VGA colour screen, all for just £1299.

☎ 0254 682 343

Corporate Sales call 0254 696 060

TIME
Computer Systems Ltd.

Time House
Devonport Road
Blackburn
Lancs. BB2 1EJ

SANYO

Specification

- 386sx-20 processors
- Whisper quiet 60Mb hard drive
- 3.5" 1.4Mb floppy drive
- 387sx maths co-processor socket
- 387sx is factory fitted as standard in M models
- 3Mb RAM expandable to 5Mb
- Superb LCD display with 32 grey scales, adjustable to any position over 135 degree angle – Bright and Clear!
- Superb external Panasync 14" SVGA colour screen option with 2 year onsite warranty – Connects to VGA port on notebook.
- Full travel responsive keyboard, one of the best amongst notebooks
- Serial and Parallel ports
- Internal slot for Data or Fax/Data modems
- Handle for easy carriage when the briefcase is not around
- Amazing battery life (upto 6 hours with power management)
- Quick battery recharge (within one hour)
- A wide range of options and accessories
- 3 year E.C. warranty

Model	NB17 286-12		NB18 386sx-20	
	20 Mb	**60 Mb**	**60 Mb**	**80 Mb**
1 Mb	£799	£899	£999	£N/A
3 Mb	£949	£1049	£1149	£1449
5 Mb	£1099	£1199	£1299	£1599

386-60M and 386-80M Models include a built-in Maths Co-processor & Ball-point Mouse: only **£100** extra

Please call our staff for details of options. Above prices do not include Panasync screen.

386sx-20 60Mb 3Mb £1299

With 3 year warranty and external 14" Panasync SVGA colour screen

Sales lines open till 7pm. Technical/help line: (0254) 682 343. (Mon-Fri 9-5). **Courier Delivery**: PC's £15. Visa/Access, Government, Educational and Corporate orders welcome. Leasing terms available. All prices exclude Delivery and VAT (17.5%). Prices and specs subject to change and subject to conditions of sale copies of which are available on request. All trademarks acknowledged. Collections are strictly by appointment only. **E&OE**

SNB8A